Amendments to the Specification:

Please replace paragraph [0039] with the following amended paragraph:

[0039] In yet other embodiments, portions of the video controller may be integrated into several components of the system. For example, as shown in FIG. 2c, portions of the video controller, such as video circuitry 212, may be integrated with graphics source 200, while other portions of the video controller, such as multiplanar frame buffer ~~214~~ 206, may be integrated onto the image generator 210. In the embodiment depicted in FIG. 2c, frame buffer ~~214~~ 206 and video circuitry 212 may be interconnected by a high transfer rate connection 225. Connection 225 may be any suitable connection capable of sustaining throughput sufficient to provide a flicker free image generated by image generator 210. In some embodiments, for example, this may require data (whether compressed or not) to be transmitted up to fifty times per second from the frame buffer to the image generator.

Please replace paragraph [0064] with the following amended paragraph:

[0064] Memory locations in the multiplanar frame buffer may therefore be assigned in such embodiments using the following equation:

$$Addr=N_{B/P}*(r*\text{cosine}\ (\theta)+[[N_x]]\underline{N_r}*y'+[[N_x]]\underline{N_r}*[[N_y]]\underline{N_{y'}}*r*\text{sine}\ (\theta))$$

Please replace paragraph [0068] with the following amended paragraph:

[0068] FIG. 5B shows a method of multiplanar anti-aliasing in accordance with embodiments of the invention is shown. At step 530, the fractional value of the z scaled value (z_f) is read. At steps 535 and 540, the z_f value is used to calculate two color values, RGB_{near} and RGB_{far}. The RGB_{near} value is calculated by multiplying the RGB value for that pixel by $(1-z_f)$, while the RGB_{far} value is calculated by multiplying the RGB value for that pixel by z_f. In steps 545 and 550, RGB_{near} and RGB_{far} are assigned to the frame buffer sections corresponding to z_i and z_{i+1}, respectively. Other such methods for accomplishing multiplanar antialiasing are described in U.S. application Ser. No. 09/291,315, filed on Apr. 14, 1999, now U.S. Pat. No. 6,377,229 (hereinafter "the '229[[_____]] patent") which is hereby incorporated by reference in its entirety.

Please replace paragraph [0077] with the following amended paragraph:

[0077] FIG. 5c shows a method of assigning memory locations in the multiplanar frame buffer using the multi-pass rendering technique. At step 530, a pair of software clip planes is defined. The software clip planes bound the part of the 3D scene to be displayed on a single optical element, making the rest of the 3D scene invisible. Once the clip planes are defined, the 3D scene is rendered at step 540 into a two dimensional frame buffer. At step 545, anti-aliasing is preferably performed. Anti-aliasing can be accomplished using any suitable function, such as, for example, a fog function. Such functions are described in, for example, in the '229[[_____]] patent. At step 550, the rendered data is read out to the frame buffer without the need for further processing.

Please replace paragraph [0082] with the following amended paragraph:

[0082] FIG. 5d shows an embodiment of a direct rasterization process in accordance with the invention. At step 570, fragment information is computed. As discussed above, fragment information may include the RGB values of each pixel defined by a polygon primitive. As also discussed above, the fragment information is computed as the pixel data is being processed from the vertex information of the surface primitives.

At step 572, a memory location in the multiplanar frame buffer is computed for the computed fragment. This can be accomplished in any suitable manner in which the x, y, and z coordinates are used at least in part to determine the memory address. In some embodiments, for example, where the planar resolution of the 3D image is 640 pixels by 480 pixels, the address may be calculated using either of the following equations:

B5

$$Addr=N_{b/p}*(x+640*y+640*480*[[z_1]]\underline{z_i})$$

$$Addr=N_{B/P}*(r*\text{cosine}\ (\theta)+[[N_x]]\underline{N_r}*y'+[[N_x]]\underline{N_r}*[[N_Y]]\underline{N_y}*r*\text{sine}\ (\theta))$$

Please replace paragraph [0084] with the following amended paragraph:

[0084] At step 574, depth testing is performed. If a pixel is located behind (i.e., at higher z) a previously processed opaque pixel, the new pixel is not visible and the corresponding pixel data can be discarded as shown in step 582. If the pixel is not located behind an opaque pixel, the new pixel is visible and the corresponding pixel data is retained for further processing.

B6

Please replace paragraph [0085] with the following amended paragraph:

[0085] At step 57[[4]]6, antialiasing is performed on the image fragment. In an embodiment, the fractional portion of the z value is used to calculate RGB values of adjacent pixels. An example of such a function is disclosed in the '229[[_____]] patent, and as discussed above. Any suitable antialiasing function, however, may be used.

Please replace paragraph [0086] with the following amended paragraph:

[0086] Optionally, at step 57[[6]]8, alpha blending may be performed on the multiplanar image. Any suitable alpha blending technique may be used, such as, for example, conventional alpha blending as discussed above. Alternatively, in a multiplanar frame buffer, the brightness and/or color of pixels located behind (i.e., at the same x,y location but at a larger z) a new pixel may be modified if it can be viewed through the new foreground pixel (i.e., the new foreground pixel is translucent). For example, background pixels may have their brightness and/or color modulated depending on the alpha value of the foreground pixel.